

УДК 621.039.564

АВТОМАТНОЕ ПРОГРАММИРОВАНИЕ ПЛИС В ЗАДАЧАХ УПРАВЛЕНИЯ ЭЛЕКТРОПРИВОДОМ

Ю. Ю. Янкин,

аспирант

ОАО «Концерн «НПО «Аврора»

А. А. Шалыто,

доктор техн. наук, профессор

Национальный исследовательский университет информационных технологий, механики и оптики

Рассматривается применение автоматного программирования для реализации блоков управления электроприводом, построенных на основе микросхем программируемой логики. Показана эффективность такого подхода. Приведены результаты испытаний блоков.

Ключевые слова — автоматное программирование, конечный автомат, электропривод, программируемые логические интегральные схемы.

Введение

В системах управления техническими средствами кораблей и судов часто возникает необходимость регулирования положения того или иного исполнительного органа. Для решения таких задач могут использоваться различные типы приводов — электромеханический, гидравлический, пневматический и т. д. Каждый из них имеет свои особенности и область применения. В настоящей работе рассматриваются блоки управления электромеханическим приводом на основе специально разработанного для таких электроприводов шагового двигателя (ШД). Указанные блоки могут быть построены на основе аналоговых микросхем, цифровых микросхем низкой степени интеграции, цифровых микросхем высокой степени интеграции или микроконтроллеров. Настоящая работа посвящена автоматной реализации блоков управления электроприводом на основе программируемых логических интегральных микросхем (ПЛИС).

Традиционно их программирование осуществляется с использованием языков описания аппаратуры (Verilog, VHDL), а также функциональных схем. Предлагаемый подход отличается тем, что алгоритм управления электроприводом задается в общем случае в виде системы графов переходов конечных автоматов, как это было предложено в работе [1]. Текст на языке описания аппа-

ратуры, необходимый для синтеза устройства в САПР производителя микросхем программируемой логики, формируется автоматически средствами пакета MATLAB. Предлагаемый подход является развитием автоматного программирования [2] применительно к программированию аппаратуры. Вопрос о применении автоматов при создании программ управления электроприводом для микроконтроллеров описан в работе [3].

Предлагаемый подход

Подход состоит из следующих этапов.

1. Создание схемы связей блока управления с объектом управления и системой верхнего уровня.
2. Разработка перечня и описания входных и выходных переменных.
3. Получение алгоритма работы от Заказчика (в виде словесного описания и временных диаграмм).
4. Эвристическое проектирование системы графов переходов конечных автоматов.
5. Отображение графов переходов с использованием пакета Stateflow, входящего в состав MATLAB, по методике, изложенной в работе [4].
6. Разработка модели объекта управления средствами MATLAB-Simulink.
7. Комплексное моделирование системы автоматов и объекта управления с получением временных диаграмм работы.

8. Сравнение результатов моделирования с требуемым алгоритмом работы. Если результаты моделирования не удовлетворяют требованиям Заказчика, то необходимо вернуться к этапу проектирования графов переходов.

9. Реализация системы графов переходов, представленных в Stateflow на языке описания аппаратуры (Verilog, VHDL), с помощью интерпретатора пакета HDL Coder, входящего в MATLAB.

10. Моделирование полученной программы с использованием САПР производителя ПЛИС или другой системы HDL-моделирования, например ModelSim.

11. Компиляция и последующая загрузка в целевую аппаратуру.

При разработке с использованием автоматов отладка либо занимает меньше времени, чем при традиционном подходе, либо она вообще отсутствует, так как автоматы обычно сразу работают в соответствии с требуемым алгоритмом. При этом отметим, что один и тот же автомат может быть использован как для получения исполняемого кода на языках программирования (C++), так и для получения описания устройства на языках описания аппаратуры, таких как, например, Verilog, VHDL.

Применение предлагаемого подхода исключает трудоемкий и обычно приводящий к большому числу ошибок процесс ручного программирования.

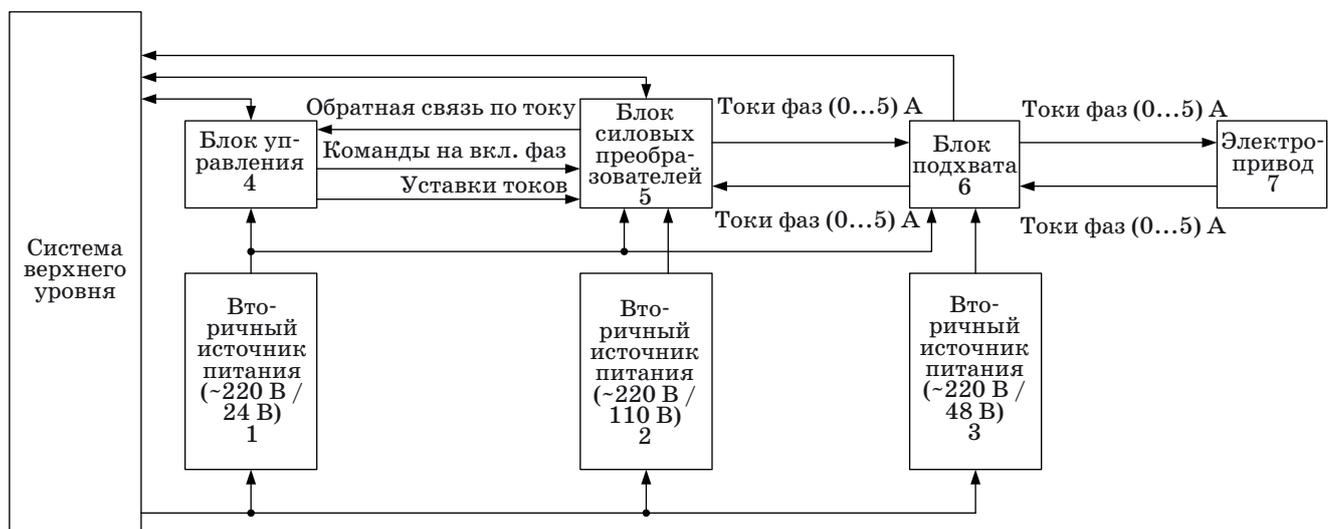
Реализации канала управления электроприводом на основе шагового двигателя с использованием предлагаемого подхода

Структурная схема канала управления электроприводом на основе четырехобмоточного ШД показана на рис. 1.

Вторичный источник питания 1 предназначен для формирования напряжения питания внутренних цепей блоков 4–6; источник 2 — для питания силового преобразователя блока 5; источник 3 — для питания силового преобразователя блока 6; блок управления 4 — для формирования режимов работы электропривода и уставок токов фаз ШД в соответствии с сигналами, поступающими из системы верхнего уровня; блок силовых преобразователей 5 — для формирования токов фаз ШД в соответствии с уставками, поступающими из блока управления, контроля их достоверности и формирования некоторых режимов работы электропривода в случае отказа блока управления, а блок подхвата 6 — для контроля токов, формируемых блоком силовых преобразователей 5, и «подхвата» токов фаз ШД в случае отказа блока 5. В настоящей работе структура канала управления и блоки типов 1–3, 6 не рассматриваются.

Блок управления 4 (рис. 2) содержит два функциональных узла, первый из которых (узел УФ1) предназначен для формирования режимов работы электропривода и уставок токов фаз, а второй (узел УК1) — для контроля исправности блока управления и блока силовых преобразователей. Для этого используется обратная связь по токам фаз ШД.

Блок силовых преобразователей 5 содержит три функциональных узла (рис. 3). Первый из них (узел УФ2) предназначен для контроля достоверности сигналов, поступающих из блока управления, и формирования некоторых режимов работы электропривода в случае отказа блока управления, второй (узел УК2) — для контроля исправности блока силовых преобразователей 5 и линий связи с ШД, а третий (узел УУ) — для формиро-



■ Рис. 1. Структурная схема канала управления

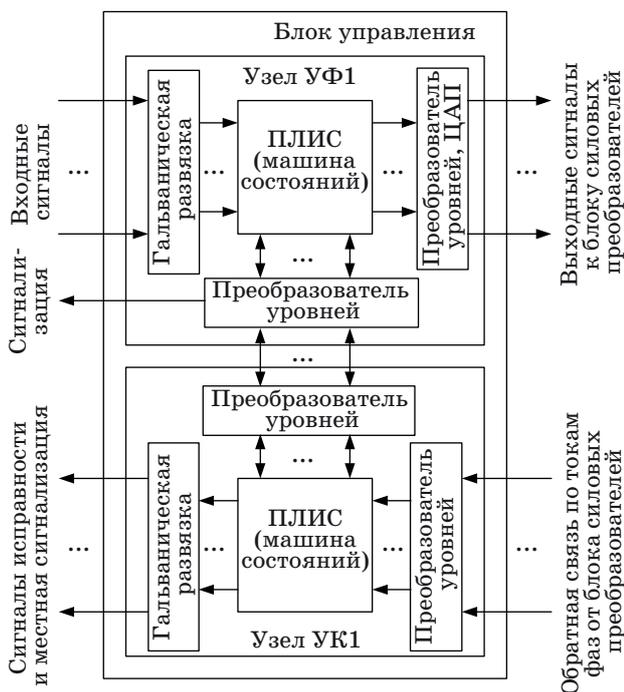


Рис. 2. Структурная схема блока управления

вания токов фаз ШД и контроля исправности силового преобразователя.

Модель фрагмента канала управления, содержащего блоки управления 4, силовых преобразователей 5 и нагрузку 7, показана на рис. 4. Модель реализована в среде MATLAB-Simulink с использованием таких его расширений как Power Systems Blockset, Stateflow, HDL Coder.

Подсистема *Inputs* на рис. 4 предназначена для моделирования части входных переменных модели блока управления 4, представленного в виде подсистемы *Type 4*. Подсистема *Type 5 and Load* содержит модель блока силовых преобразователей 5 и нагрузку 7. В качестве выходных переменных на блоки регистрации *Commands'n'References* и *Indicators* выведены следующие переменные и сигналы:

Type 4.z5, Type 4.z6 — внутренние переменные блока управления, определяющие направление движения;

CF1&FCF1...CF4&FCF4 — дискретные сигналы на включение фаз и дискретные сигналы обратной связи по току;

RF1&CuF1...RF4&CuF4 — уставки и токи фаз;

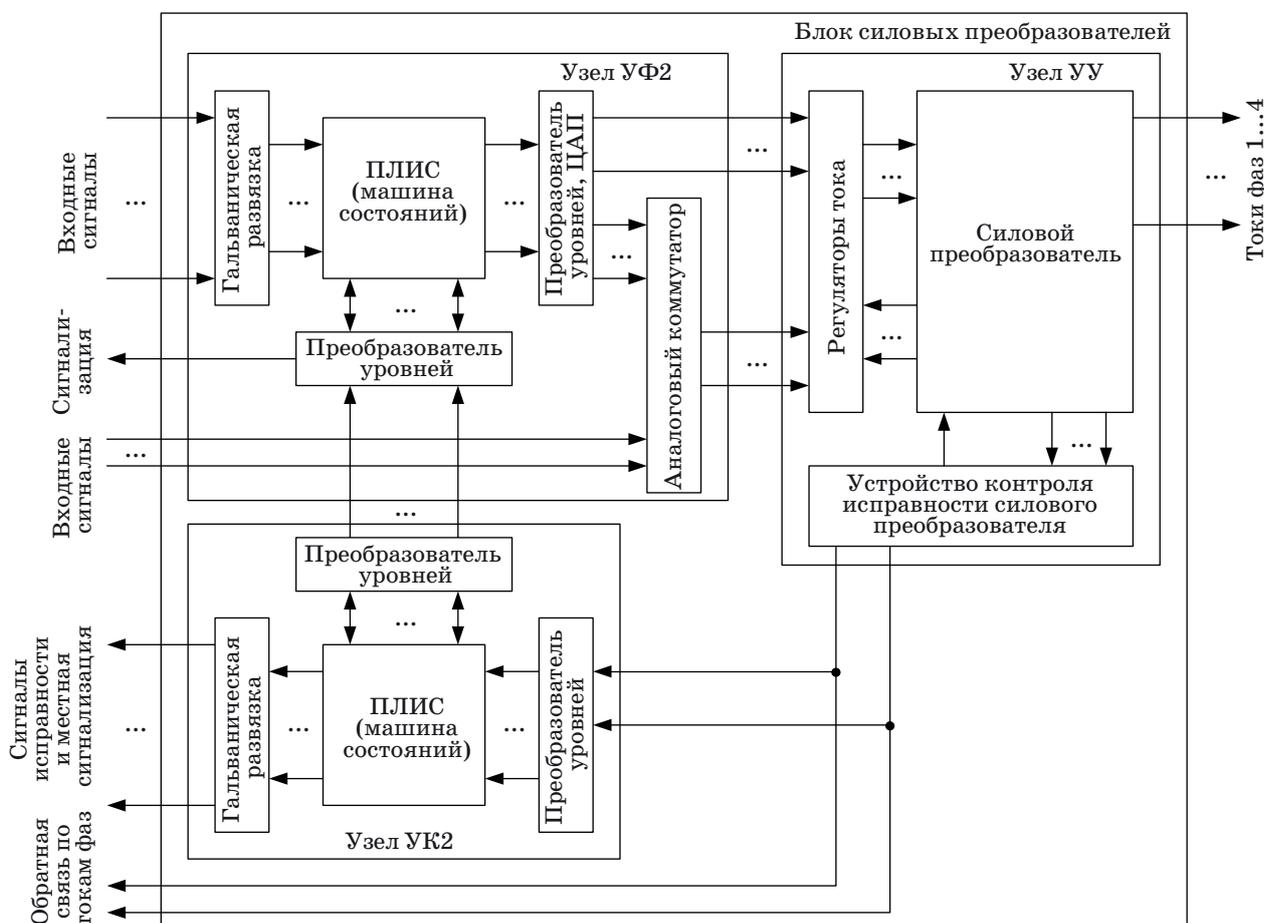
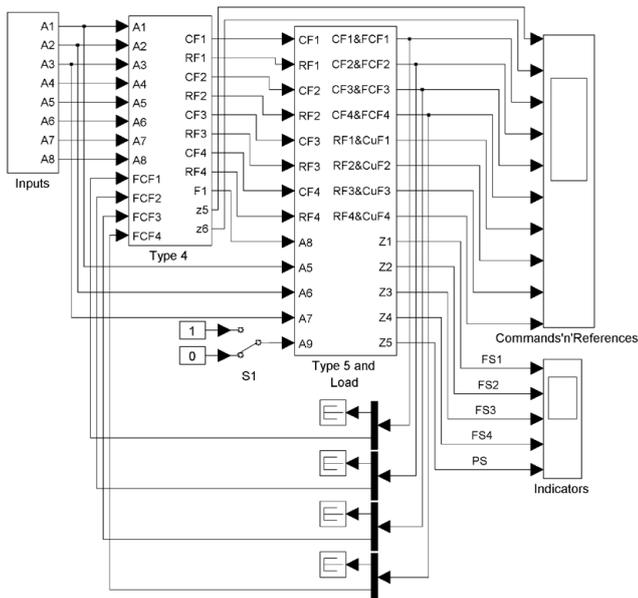


Рис. 3. Структурная схема блока силовых преобразователей



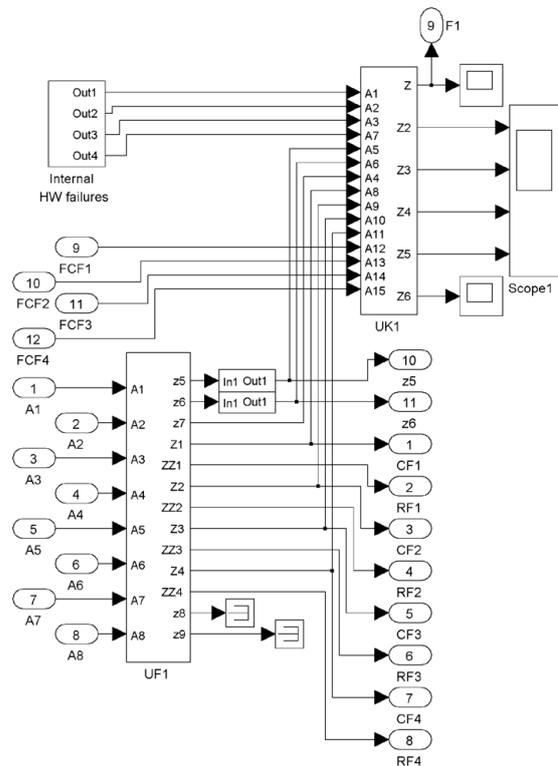
■ Рис. 4. Модель фрагмента канала

FS1, FS2, FS3, FS4, PS — сигналы и индикация неисправности блока силовых преобразователей.

Модель подсистемы *Type 4* представлена на рис. 5.

Блок *Internal HW failures* предназначен для имитации внутренних аппаратных неисправностей блока управления 4, например отсутствия вторичного электропитания, неисправности тактового генератора и т. п. Модели поведения узлов *УФ1, УК1* реализованы в виде машин состояний *UF1, UK1* соответственно.

Подсистема *Type 5 and Load* (см. рис. 4) содержит модели поведения узлов *УФ2, УК2*, выполненные аналогично моделям *UF1, UK1*, а также модель узла *УУ*. На рис. 6 представлена модель одного из четырех каналов узла *УУ*. Она содержит модель регулятора тока с широтно-импульсным модулятором (ШИМ) — *PI&PWM1*, модель силового преобразователя *HB1*, аналоговый коммутатор *SW1*, блок согласования шага интегрирова-



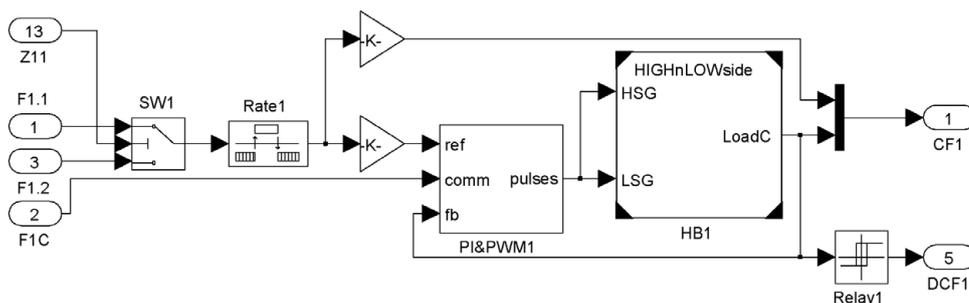
■ Рис. 5. Модель подсистемы *Type 4*

ния *Rate1* и пороговое устройство *Relay1*, предназначенное для реализации дискретной обратной связи по току.

Подсистема *PI&PWM1* (рис. 7) содержит пропорциональный интегральный регулятор тока — блок *PI* и ШИМ — блоки *SAW, CMP*, модель датчика тока *CS*. Выходными сигналами являются импульсы управления силовыми транзисторными ключами.

Модель *HB1* (рис. 8) состоит из модели источника напряжения 1 (см. рис. 1) — блок *VDC*, модели силового преобразователя *M1, M2, D1, D2, CM1* и нагрузки *RL1*.

Всего в системе содержится четыре машины состояний *UF1, UK1, UF2, UK2*, описывающие



■ Рис. 6. Модель одного канала узла *УУ*

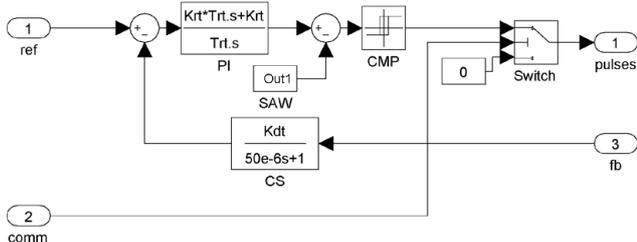


Рис. 7. Подсистема PI&PWM1

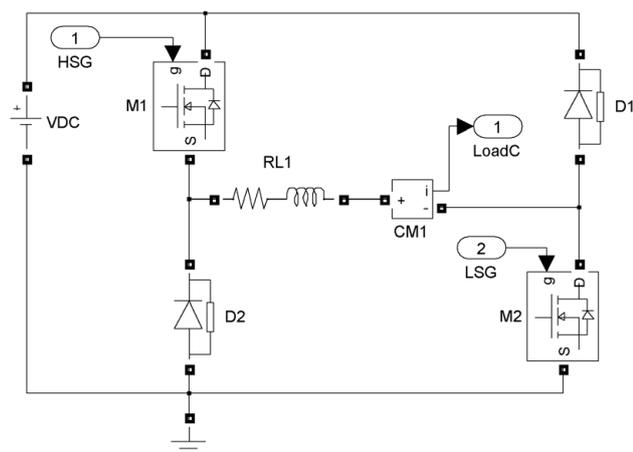


Рис. 8. Модель одного канала силового преобразователя HВ1 с нагрузкой

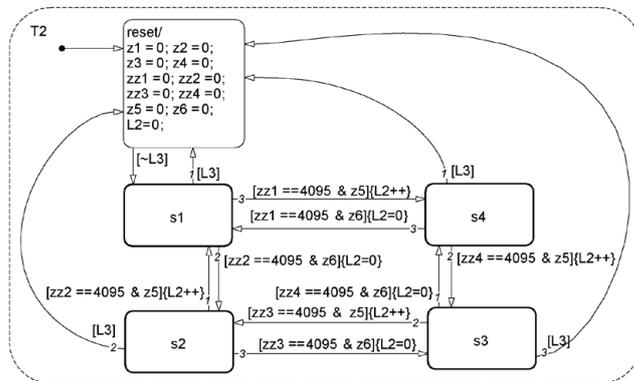
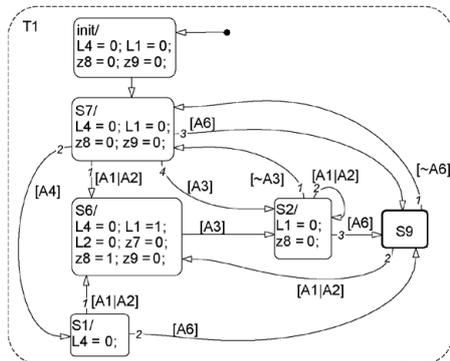


Рис. 9. Машина состояний UF1

truthtable	function C1 = f1	function C2 = f2	function C3 = f3	function C4 = f4
t = tt1(b1,b2,b3,b4,b5,b6,b7,b8)	function C5 = f5			

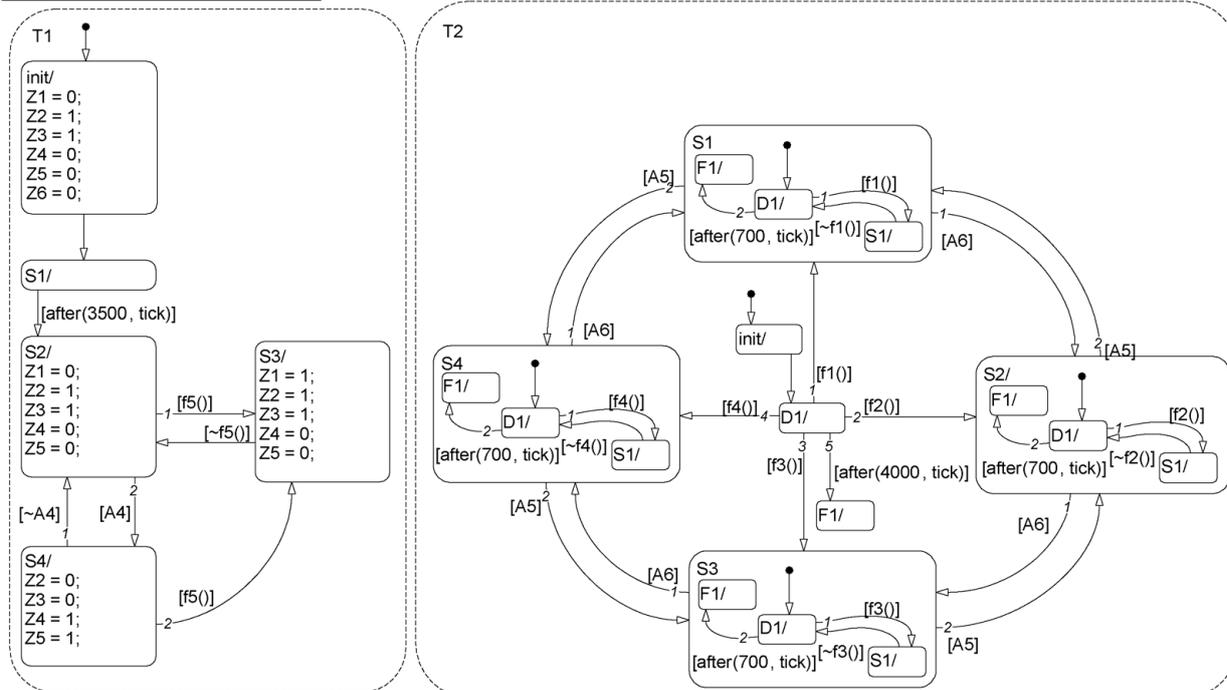
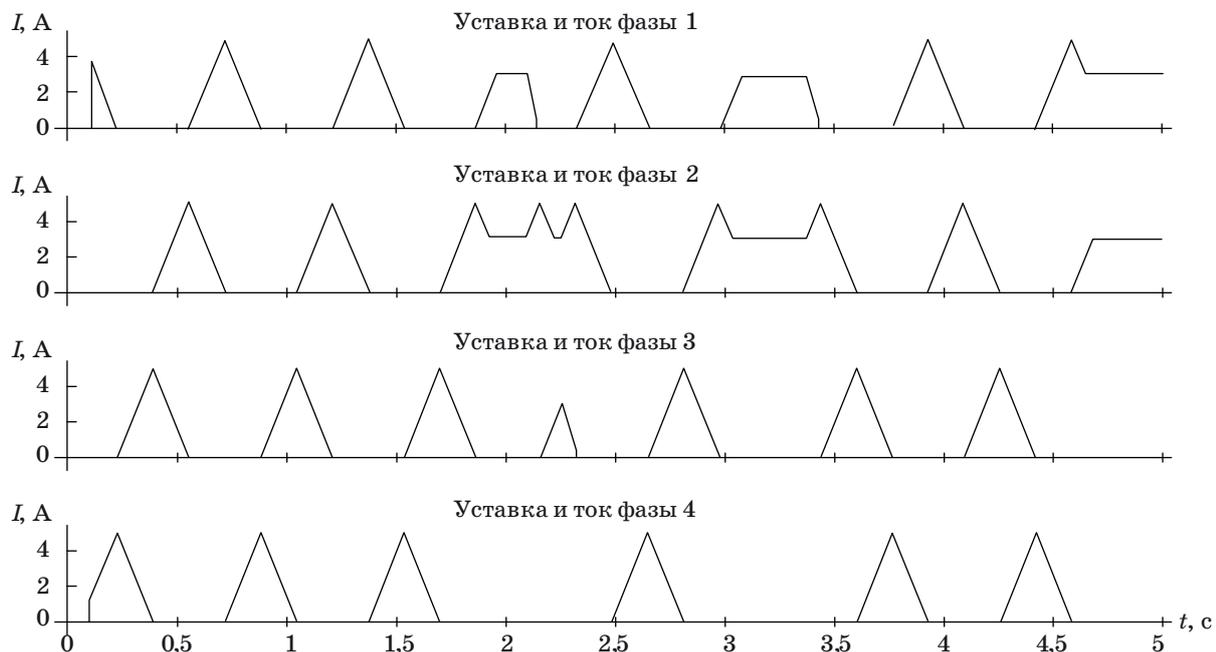
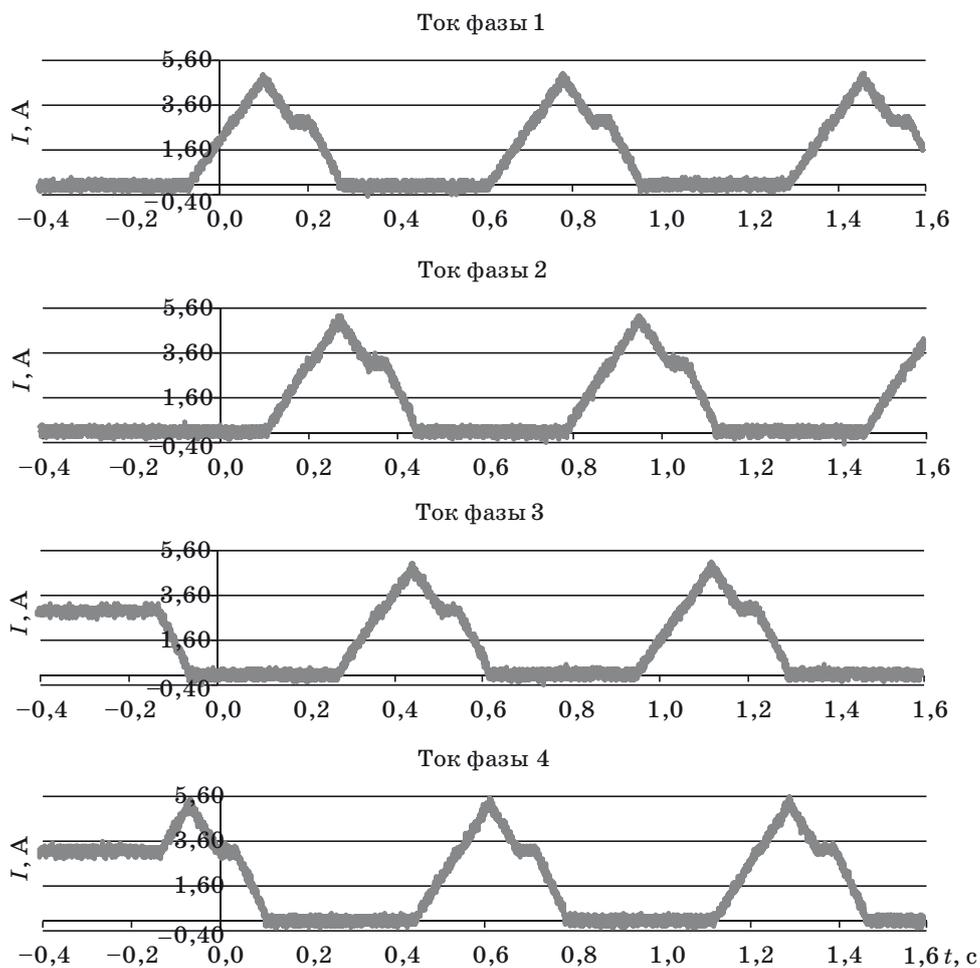


Рис. 10. Машина состояний UK1



■ Рис. 11. Результаты моделирования



■ Рис. 12. Результаты испытаний

поведение узлов УФ1, УК1, УФ2, УК2 соответственно. Машины состояний UF1 (рис. 9), UK1 (рис. 10), UK2 содержат по два гиперсостояния T1, T2, а машины узла UF2 — три гиперсостояния T1, T2, T3. При этом отметим, что гиперсостояние UF1.T1 содержит один вложенный конечный автомат S9, гиперсостояние UF1.T2 — четыре вложенных автомата S1, S2, S3, S4, гиперсостояние UF2.T2 — четыре вложенных автомата, а гиперсостояния UK1.T1, UK1.T2, UF2.T1, UF2.T3, UK2.T1, UK2.T2 не содержат вложенных автоматов. Таким образом, система состоит из 18 конечных автоматов.

Результаты моделирования в одном из режимов представлены на рис. 11.

Из их рассмотрения следует, что блоки отработали 12 шагов рабочего органа в прямом направлении, после небольшой паузы — один шаг в обратном, пять шагов — в прямом, а затем после паузы восемь шагов — в обратном и остановились в режиме удержания на фазах 1 и 2.

Результаты испытаний

Испытания опытного образца проводились в условиях стенда, блоки были установлены в штатный прибор, использовались штатные источники вторичного электропитания. В качестве нагрузки для модулей был использован штатный электропривод с нагрузочным устройством, обеспечивающим номинальную нагрузку на валу двигателя.

Из рассмотрения результатов испытаний (рис. 12) следует, что в начальный момент времени модули находились в режиме удержания на фазах 3 и 4, а затем они отрабатывают отдельные

шаги с выходом на ток удержания после окончания каждого шага; такой режим работы определяется входными сигналами из системы верхнего уровня. Сравнение графиков на рис. 11 и 12 показывает, что результаты моделирования и испытаний практически совпадают.

В заключение раздела отметим, что, несмотря на наличие достаточно большого числа автоматов в системе, особых проблем с отладкой не было. Это объясняется такими особенностями структуры автоматной части системы, как иерархичность, параллельность автоматов и состояний, а также вложенностью.

В ходе проектирования и испытаний система существенно эволюционировала, начав с шести автоматов. При этом каждое расширение ее функциональности происходило достаточно просто. Поэтому излагаемый подход может быть назван *ориентированным на внесение изменений*.

Заключение

Применение автоматного программирования по сравнению с традиционным подходом, состоящим в непосредственном программировании на языке описания аппаратуры, является более наглядным, доступным для интуитивного понимания, хорошо документируется, легко модернизируется человеком, который даже не принимал участия в разработке. Процесс программирования ПЛИС с применением автоматов формализован, что облегчает его использование и уменьшает число ошибок. При таком подходе возможно также применение методов формальной верификации программного обеспечения.

Литература

1. Harel D. Statecharts: A Visual Formalism for Complex Systems // Science of Computer Programming. 1987. Vol. 8. P. 231–274.
2. Поликарпова Н. И., Шалыто А. А. Автоматное программирование. — СПб.: Питер, 2010. — 176 с.
3. Козаченко В. Ф. Эффективный метод программной реализации дискретных управляющих автоматов во встроенных системах управления / Элект-
4. Basic Workflow for Building a Stateflow Chart / R2010b MathWorks Documentation. <http://www.mathworks.com/help/toolbox/stateflow/gs/bqnmvk8.html> (дата обращения: 26.10.2010).

ропривод и автоматика. http://www.motorcontrol.ru/publications/state_machine.pdf (дата обращения: 25.10.2010).